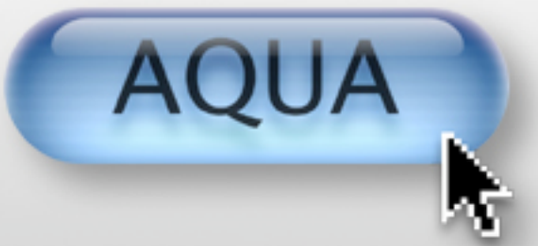
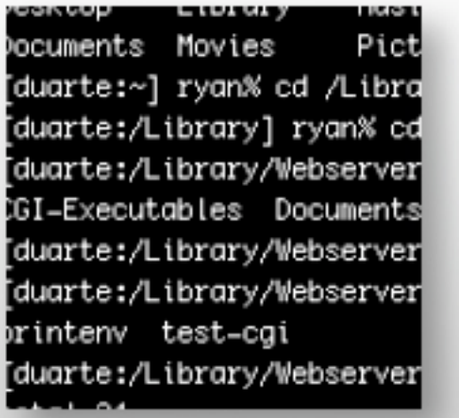
Mac OS X Accessibility

Session 421





Mac OS X Accessibility

Travis W. Brown
Technology Evangelist
Worldwide Developer Relations

Agenda

- Accessibility overview
- Mac OS X's approach to accessibility
  - Architecture
  - Universal access
  - Mac OS X Accessibility API
- Supporting the Accessibility API
- How to Bring Assistive Technology to "Panther"
  - Choosing the right starting point
  - Leveraging Mac OS X technologies

Apple Is Committed to Making Its Products Accessible to All Users

Accessibility Is Important!

- 54 million* people in the US have disabilities
  - These are our mutual customers!
- Accessibility is a sales requirement
  - Federal, State, and Local
  - Education
  - Section 504 and 508
- Right thing to do!

* U.S. Department of Education

A Quick Note on Section 508

- In 1998, the federal government re-authorized Section 508 of the Rehabilitation Act
  - Sets down minimum accessibility requirements for operating systems, applications, websites and electronic devices
  - In effect as of June 21, 2001
- Section 508 is “THE” Accessibility standard

Section 508

- Visit the Section 508 website:
  - www.access-board.gov/508.htm
  - Tip: Read the FAQ & commentary documents
- Don’t be discouraged!
  - OS Vendors and Assistive Technology play an important role
  - We can help you make the right connections

Mac OS X's Approach to Accessibility

Accessibility In Two Parts

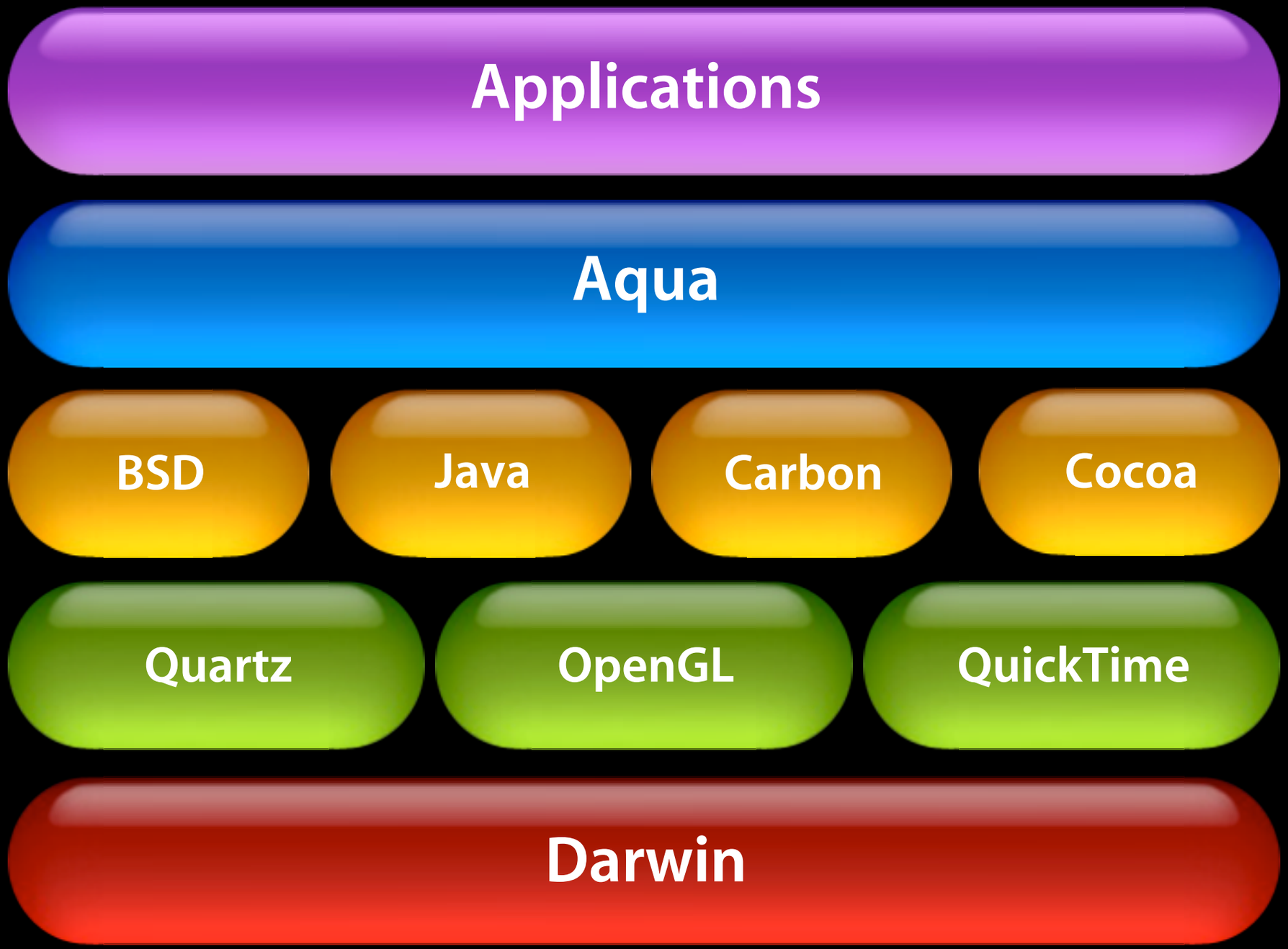
- User-Level Accessibility
  - "Built In" functionality
  - Universal Access features
- Developer-Level Accessibility
  - OS infrastruction to support 3rd party assistive technology developers
  - Mac OS X Accessibility API

Universal Access

Universal Access Tour

Travis W. Brown
Technology Evangelist
Worldwide Developer Relations

Completely New Architecture

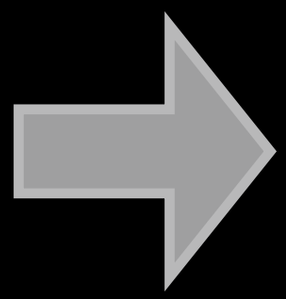


Challenges for the Community

- New architecture requires new approaches to accessibility
  - Protected Memory prevents one application from accessing another
  - Device driver model abstracts and hides the hardware
  - Visual environment is much more complex
- Some of the challenges are actually benefits!

An Opportunity to Do It Right

Lessons Learned the Hard Way



Assistive Devices

Working From a New Strength

BSD

Java

Carbon

Cocoa

Accessible Frameworks…

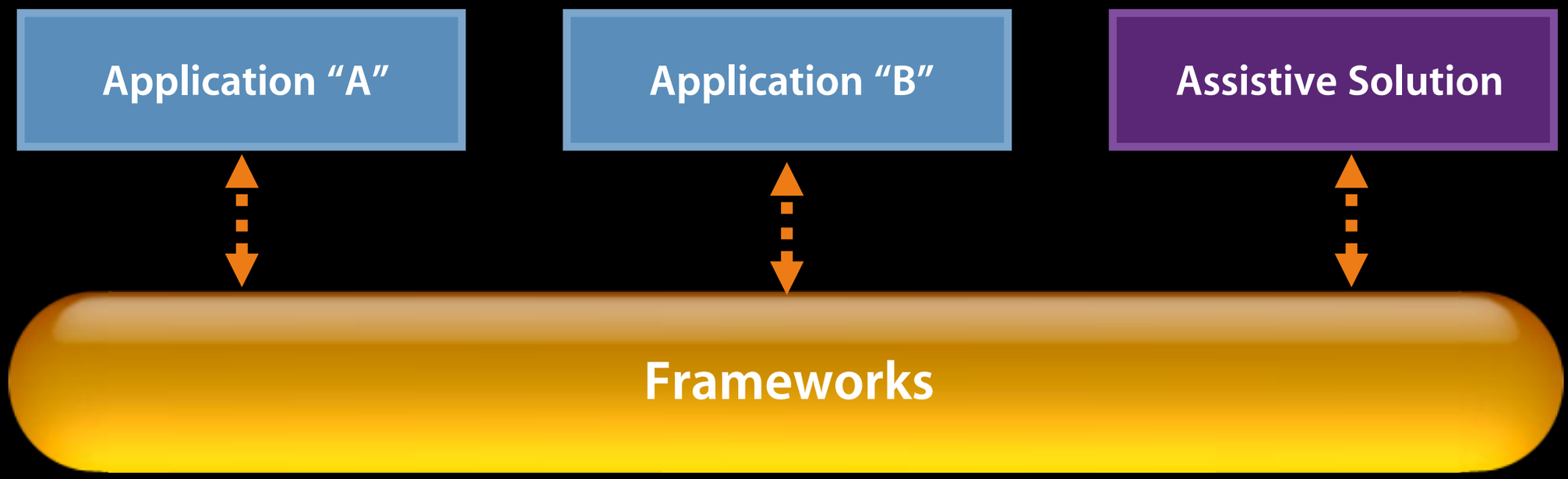
BSD

Java

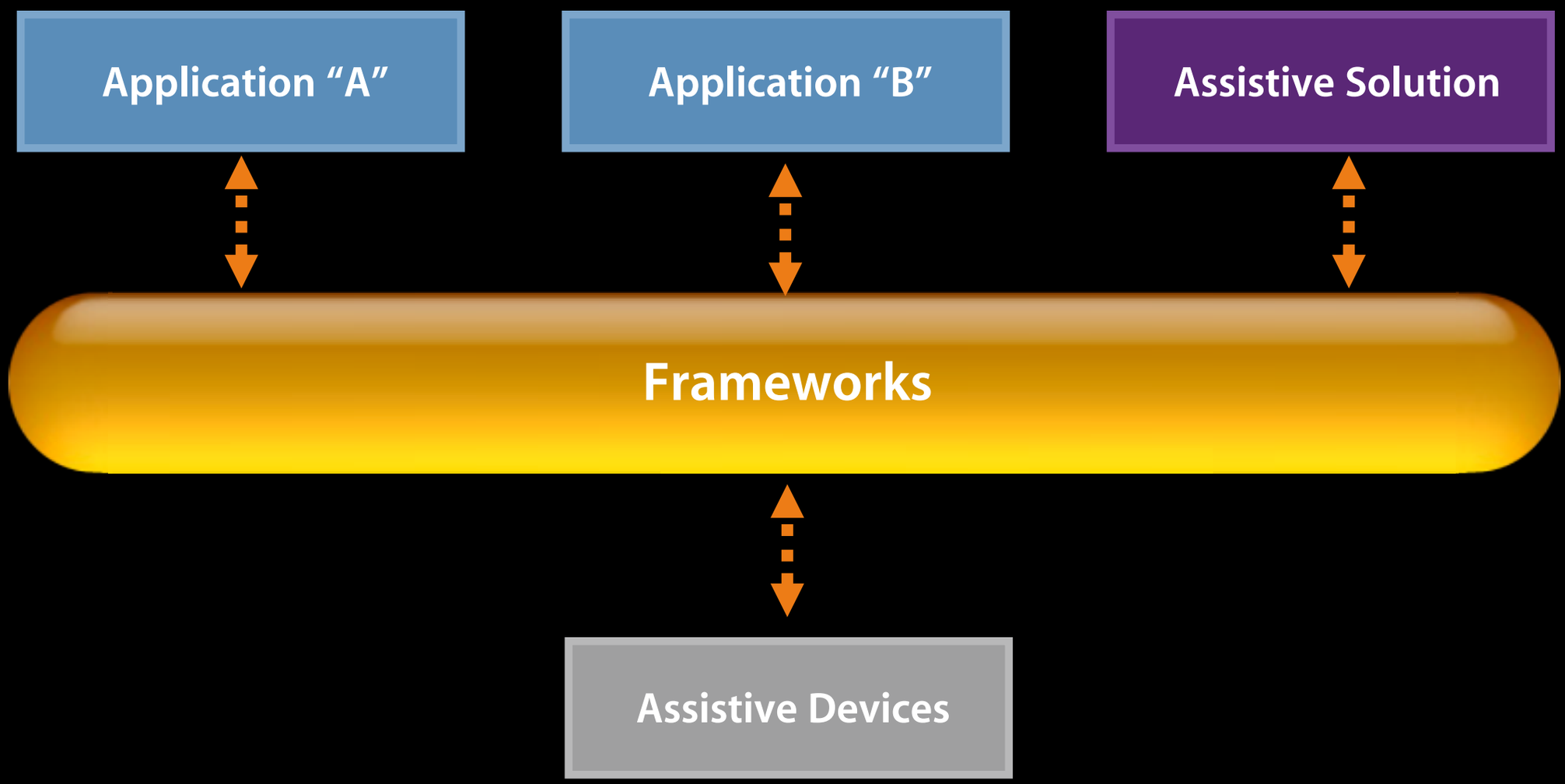
Carbon

Cocoa

Leveraging the Infrastructure



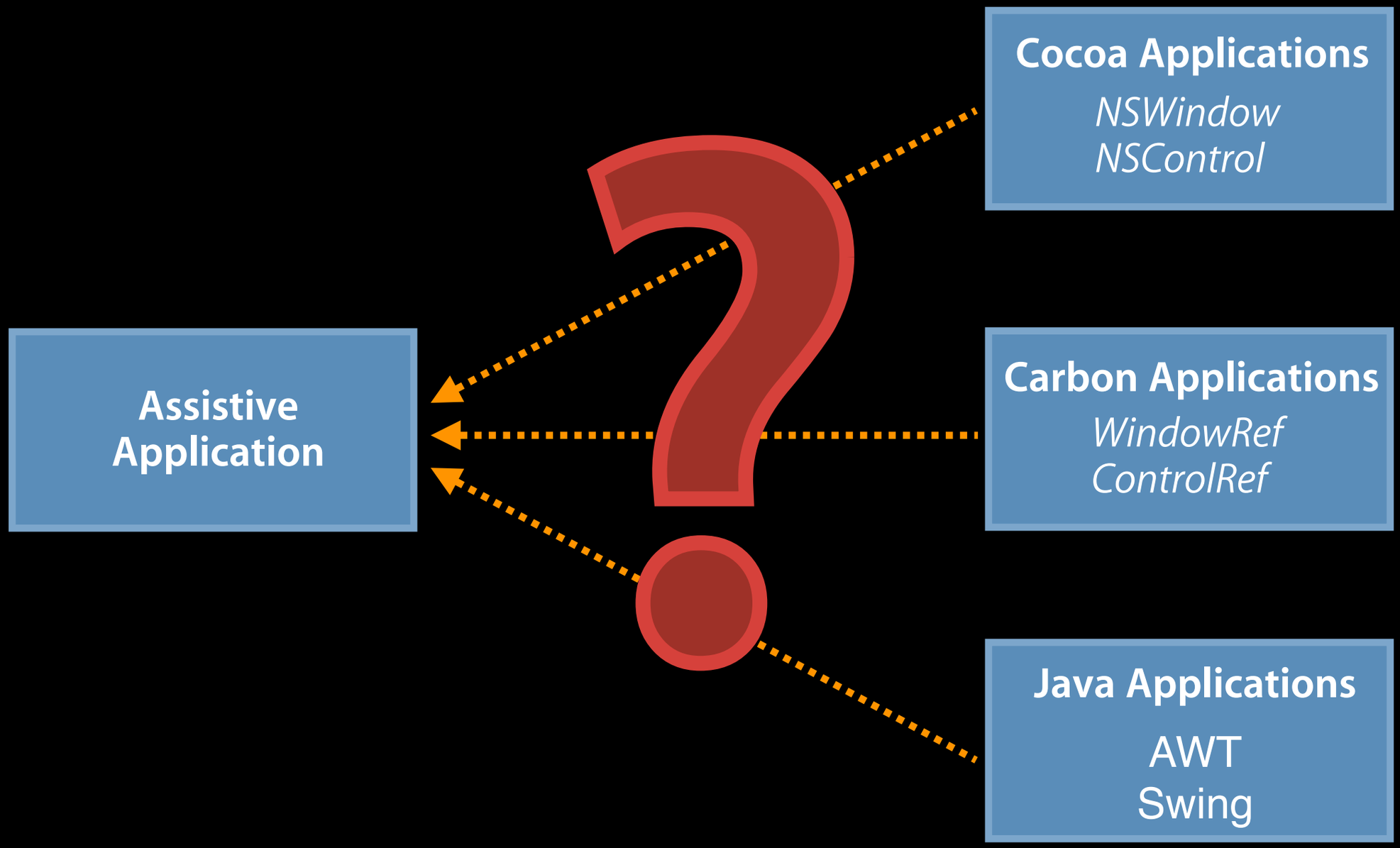
Just Another Application



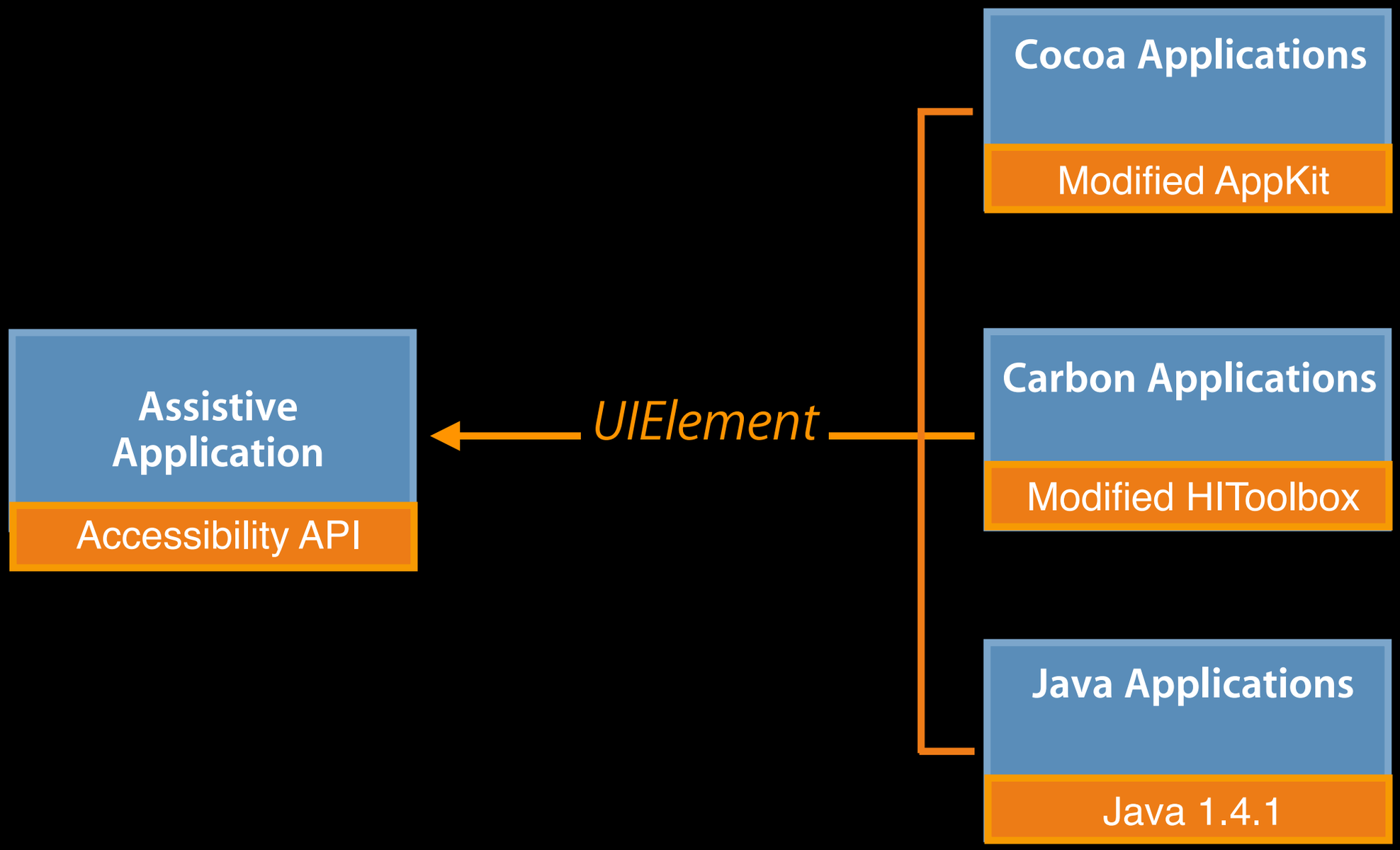
Mac OS X Accessibility API

- API-level support for Accessibility
  - Usable from Cocoa and Carbon
- Allows applications to discover and control UI in other processes
- Supplements existing APIs for HI events
- Common ground for assistive technology providers, application developers, and Apple
- Available in Mac OS X 10.2 or later
- Enhanced in “Panther”

UI Representation—Problem



UI Representation—Solution

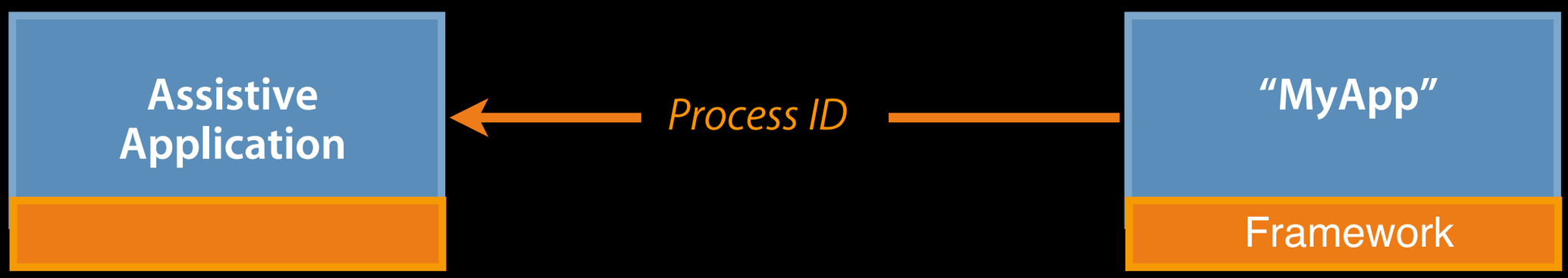


UIElements Defined

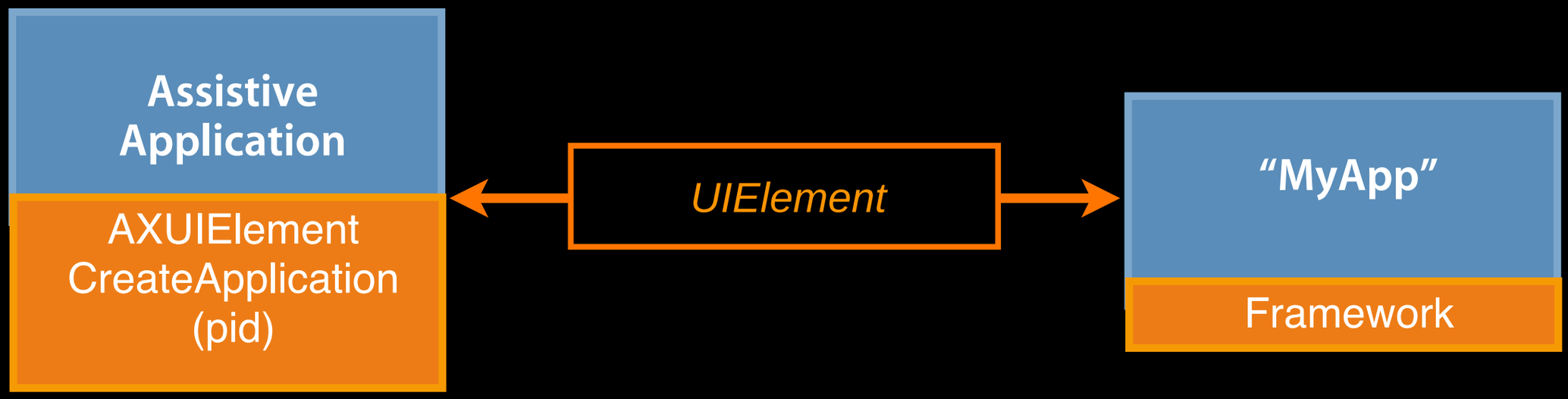
- UIElements are a framework independent abstraction for user interface representation
- Applications appear as a hierarchy of UIElements
- Attributes
  - kAXTitleAttribute
  - kAXValueAttribute
  - kAXChildrenAttribute
- Actions
  - kAXPressAction
  - kAXIncrementAction

Using the Accessibility API

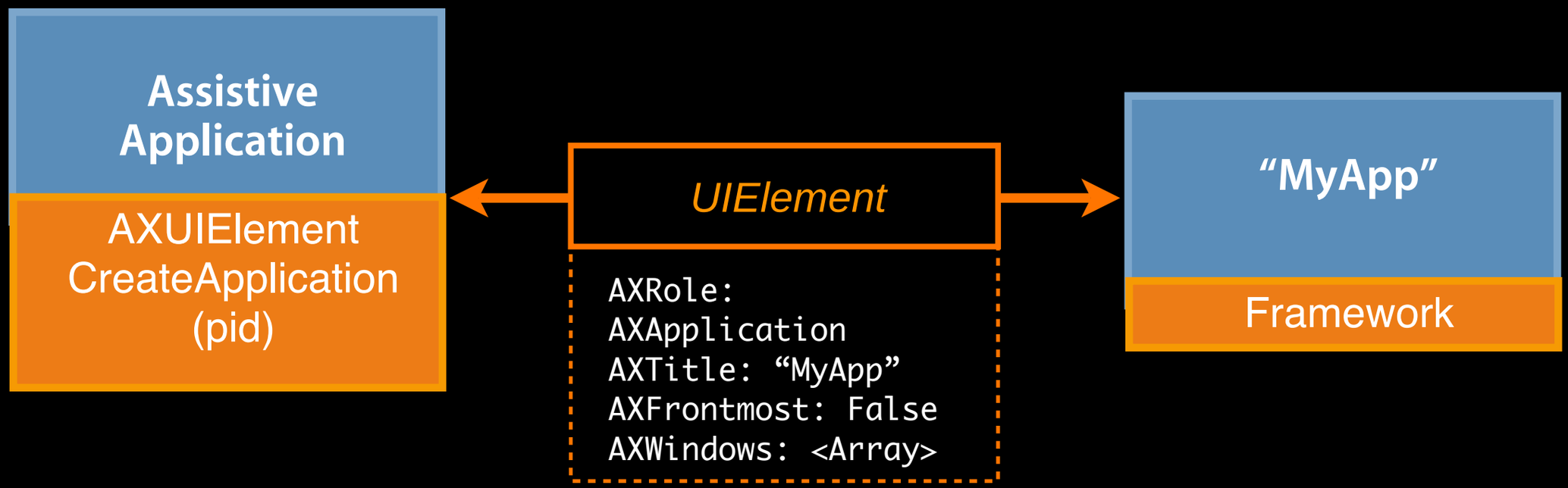
Get the PID for the Target App



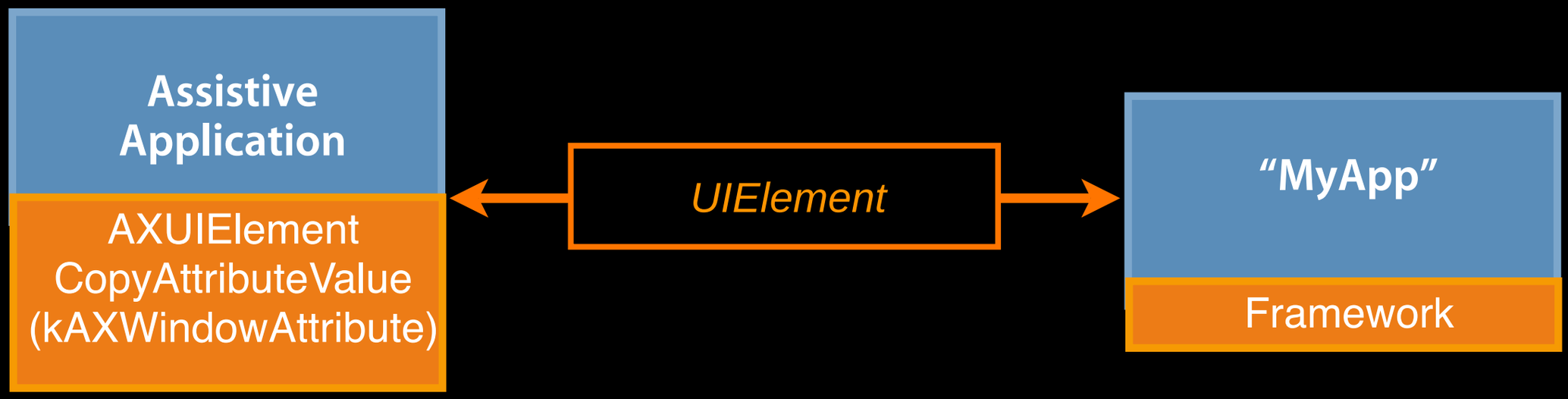
Get the Application's Info



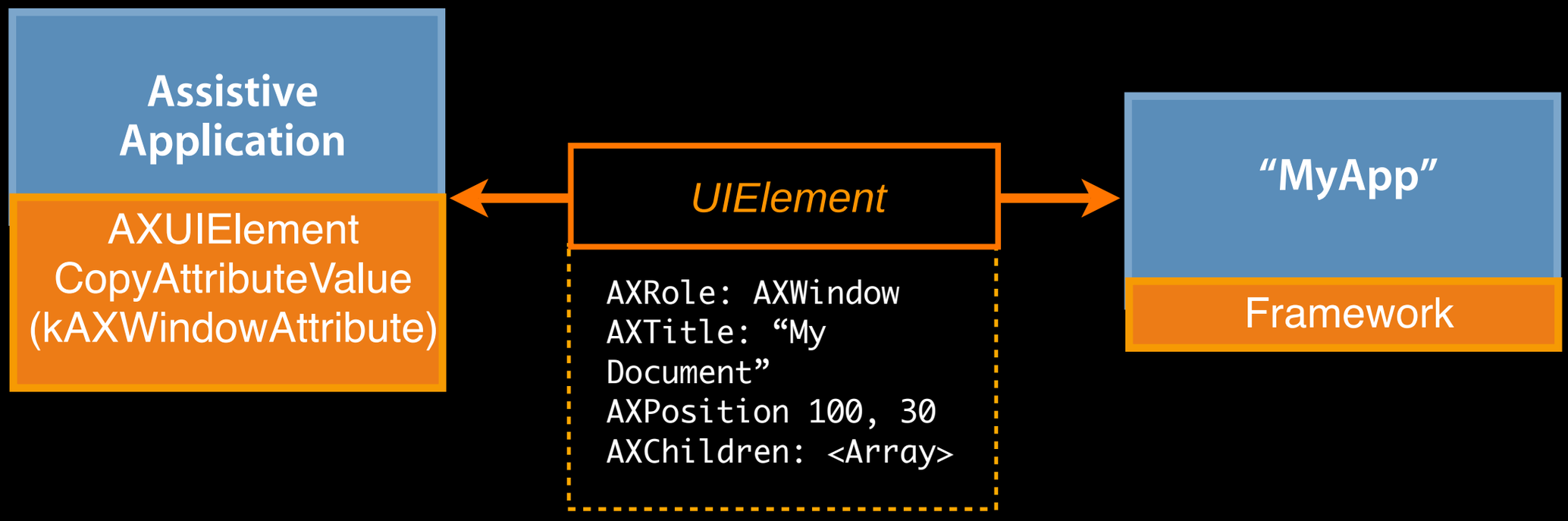
Get the Application's Info



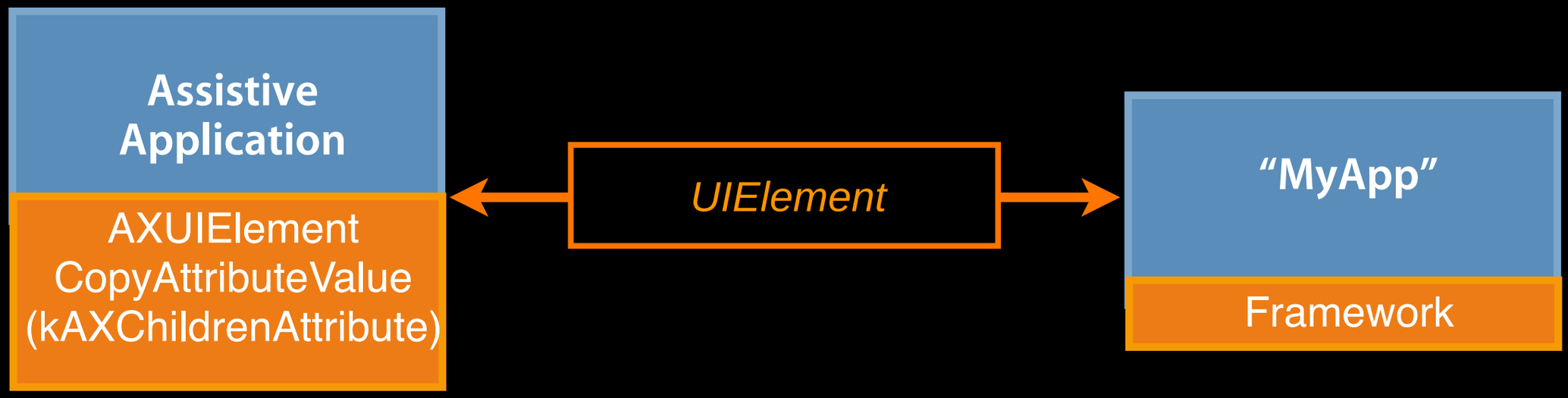
Get the Application's Windows



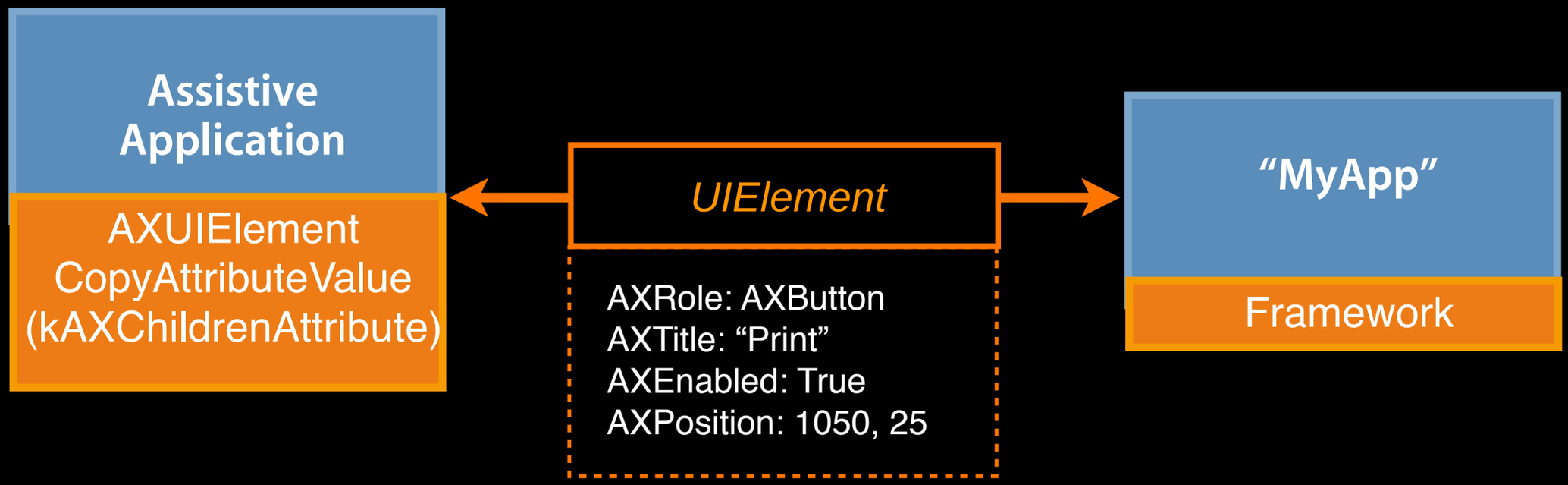
Get the Application's Windows



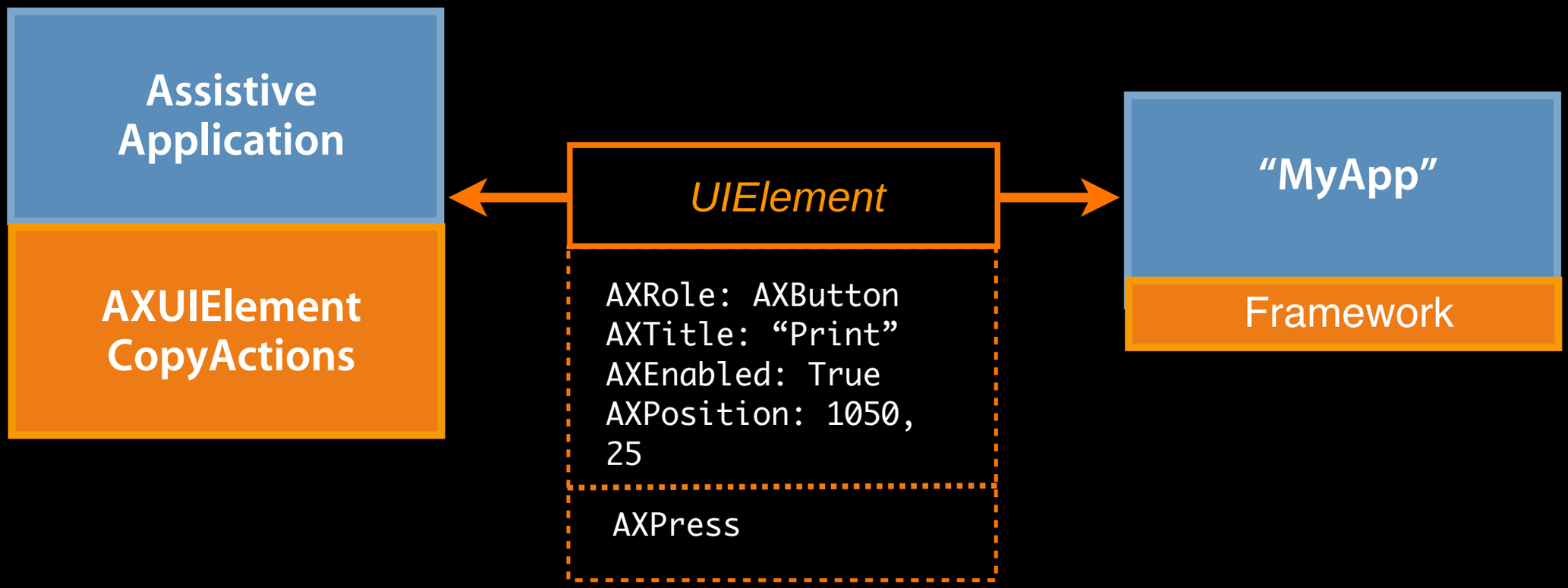
Get the Window's Children



Get the Window's Children



Get the Actions for the Controls



Send an Action

Accessibility API Notifications

- Register a callback to occur on a particular event
  - Window created, moved, miniaturized
  - Application activated/deactivated
  - UIElement destroyed
  - Value changed

Demo

UIElement Inspector

Mike Engber
job title (36 chars max)
Platform Experience Engineering

New for “Panther” Parameterized Attributes…

- Value requests require a parameter
- Character at this position
  - Specify a point
- Bounds enclosing a these characters
  - Specify a character range

Parameterized Attributes

- Two new APIs

```
AXUIElementCopyParameterizedAttributeNames (
   AXUIElementRef element,
   CFArrayRef *names)

AXUIElementCopyParameterizedAttributeValue (
   AXUIElementRef element,
   CFStringRef attribute,
   CFTypeRef parameter,
   CFTypeRef *result)
```

Parameterized Attributes—Part 3

- Text parameterized attributes

Attribute Name	Parameter Type	Value Type
kAXLineForIndexParameterizedAttribute	CFNumberRef	CFNumberRef
kAXRangeForLineParameterizedAttribute	CFNumberRef	AXValueRef-CFRange
kAXStringForRangeParameterizedAttribute	AXValueRef-CFRange	CFStringRef
kAXRangeForPositionParameterizedAttribute	AXValueRef-CGPoint	AXValueRef-CFRange
kAXRangeForIndexParameterizedAttribute	CFNumberRef	AXValueRef-CFRange
kAXBoundsForRangeParameterizedAttribute	AXValueRef-CFRange	AXValue-CGRect
kAXRTFForRangeParameterizedAttribute	AXValueRef-CFRange	CFDataRef
kAXStyleRangeForIndexParameterizedAttribute	CFNumberRef	AXValueRef-CFRange

Other Text Attributes

- New in Panther

Attribute Name	Value Type
kAXNumberOfCharactersAttribute	CFNumberRef
kAXVisibleCharactersAttribute	AXValueRef-CFRange
kAXSharedTextUIElementsAttribute	CFArray of AXUIElmentRefs
kAXSharedCharacterRangeAttribute	AXValueRef-CFRange

Window Attributes and Notifications

- New in Panther

Attribute Name	Value Type
kAXDefaultButtonAttribute	AXUIElementRef
kAXCancelButtonAttribute	AXUIElementRef

Notification Name
kAXDrawerCreatedNotification
kAXSheetCreatedNotification

Window Subroles

- New in Panther

```
kAXStandardWindowSubrole

kAXDialogSubrole

kAXSystemDialogSubrole

kAXFloatingWindowSubrole

kAXSystemFloatingWindowSubrole

kAXUnknownSubrole
```

Demo

“Monkey See, Monkey Do”

Mike Engber
job title (36 chars max)
Platform Experience Engineering

Supporting the Accessibility API

Accessible Applications…

- Carbon applications
  - Applications that use standard controls and Carbon Events receive the benefits of the Accessibility API
    - Not all Carbon applications are accessible
  - Applications with custom controls or non-standard UI behavior must be modified
  - Text engines must also be modified to support Accessibility
  - For more info see Tech Note TN2092

Accessible Applications

- Cocoa applications
  - Automatically support the Accessibility API through inheritance from UI classes
  - Completely custom UI classes can be made accessible
- Java applications
  - Java Accessibility Bridge is not required
  - Accessibility API works against AWT & Swing applications
  - Java J2SE 1.4.1 only

Bringing Your Assistive Technology to Mac OS X

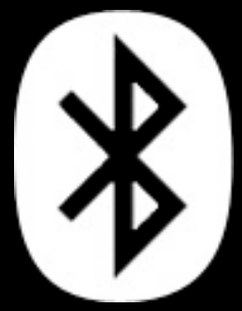
Choosing the Right Framework

- Cocoa is the right choice for most assistive technology
  - Object oriented programming model makes development efforts more efficient
  - Large number of classes provide building blocks for applications
  - Integrated tools speed development
  - Architecture allows Apple to enhance Cocoa and applications automatically inherit benefits

Choosing the Right Connections

Modern I/O for simplicity, cost, and ease of development

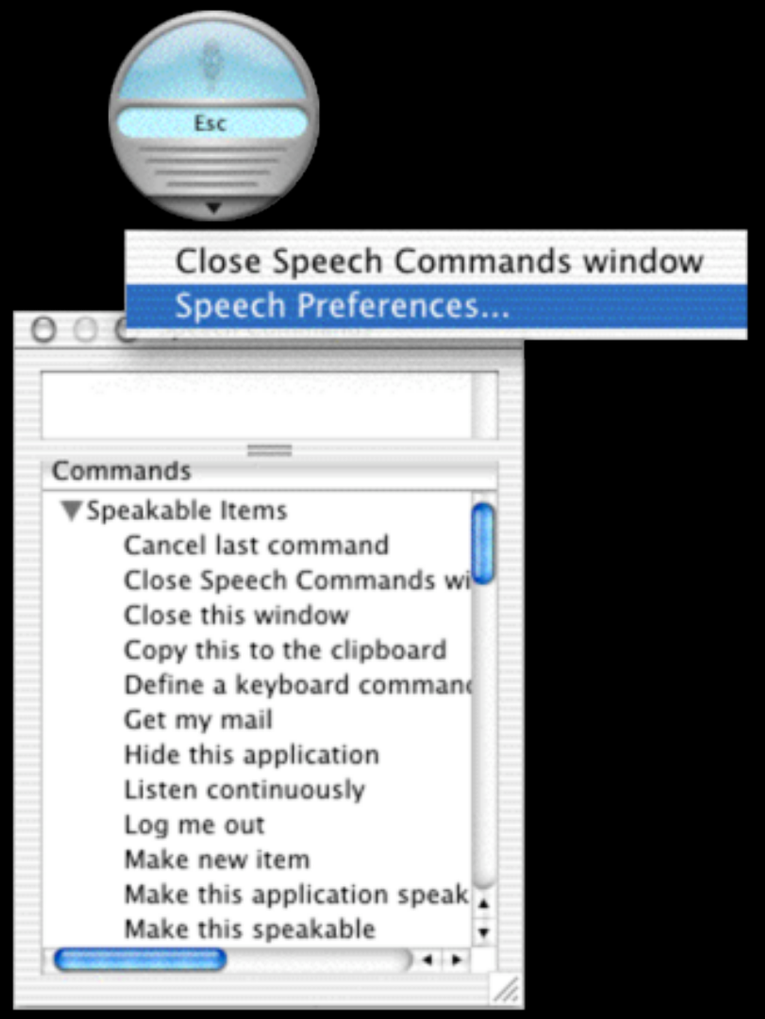
- Adopt modern serial I/O
- Follow the specifications!
  - Be careful emulating keyboards and mice
  - Check out USB HID
- Push more functionality into the software
- Interesting wireless and force feedback opportunities



Robust Speech Technology

Built in speech synthesis and speech recognition

- Speech Synthesis engine
  - Full API for conversion of text into speech
- Speaker independent recognition engine
  - Over 100,000 word vocabulary
  - Great for command and control



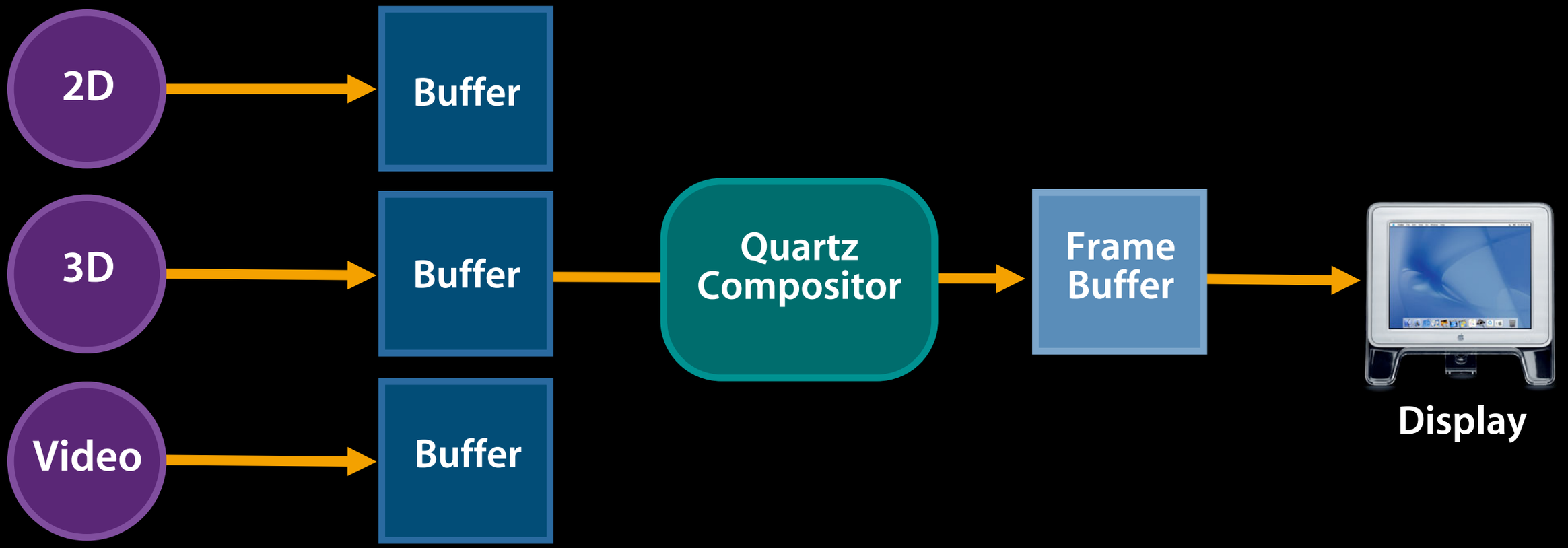
Demo

Speech

Kim Silverman, PhD
Principal Research Scientist
Manager, Spoken Language Technologies

Powerful “Visual Pipeline”

Fully integrated graphics and windowing system



“Visual Pipeline” Advantages

- Windowing system supports Z-ordering
  - Supports windows that always need to be “frontmost”
  - No need to override application, drawing API or driver behavior
- Compositing window model blends 2D, 3D and multimedia/video
  - New opportunities for Assistive Technology

Demo
“Scanner”

Travis W. Brown
Technology Evangelist
Worldwide Developer Relations

Accessible Full Screen Applications

- A compatible suggestion for full screen application developers
  - Hide the menu bar
  - Hide the Dock
  - Bring up a screen sized “borderless” window at screen saver level +1
  - Defined in CGWindowLevel.h

Wrap Up

Who to Contact

Travis Brown
Technology Manager
travis@apple.com

Mary Beth Janes
Assistive Technology Partnership Manager
mjanes@apple.com

http://developer.apple.com/wwdc2003/urls.html

For More Information

Reference Library

Documentation

Aqua Human Interface Guidelines, Cocoa Programming Topic: Accessibility, Making your Carbon Application Accessible to Users with Disabilities

ADC Home > Documentation

http://developer.apple.com/documentation/Accessibility/Accessibility.html

Accessibility Reference for Assistive Applications

Mac OS X Developer Install

/Developer/Documentation/ReleaseNotes/AssistiveAPI.html

Reference Library

Documentation (Cont.)

Accessibility Overview (WWDC 2002 Session 009)

ADC Member Site > ADC TV
http://connect.apple.com/

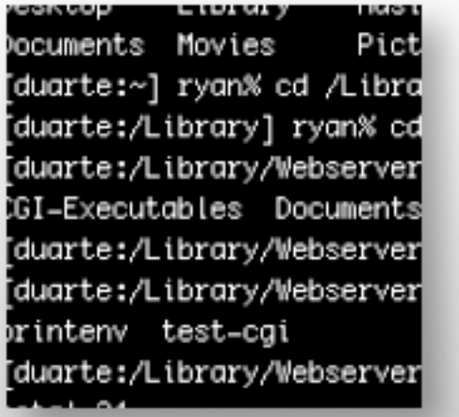
UIElementInspector Sample Code

ADC Home > Sample Code
http://developer.apple.com/samplecode/Sample_Code/OS_Utilities/UIElementInspector.htm

accessibility-dev Mailing List

http://www.lists.apple.com/mailman/listinfo/accessibility-dev

Q&A



http://developer.apple.com/wwdc2003/urls.html

WWDC 2003

Worldwide Developers Conference